UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/676,387	09/30/2003	Gabriel Monaton	15437-0635	9150

7590 03/21/2007

Bobby K. Truong
HICKMAN PALERMO TRUONG & BECKER LLP
2055 Gateway Place Suite 550
San Jose, CA 95110-1089

EXAMINER	
WANG, BEN C	
ART UNIT	**PAPER NUMBER**
2192	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/21/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	10/676,387	MONATON ET AL.	
	Examiner	Art Unit	
	Ben C. Wang	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *30 September 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-34* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-34* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-34 are pending in this application and presented for examination.

Specification Objections

2. The specification is objected to because the following informalities:

- "Ultra-Sparc", cited in P. 6, Line 18, "Sun Microsystems", in P. 9, line 19, "SPARC", cited in P. 9, Line 4, "Solaris", cited in P. 9, Line 11, are registered trademarks.
- Acronyms of "SCM", cited in P. 11, Line 11 doesn't have description for it
- "include several packages and patches (e.g., form Solaris add-on products)", cited in P. 13, line 8, should be corrected as "include several packages and patches (e.g., from Solaris add-on products)".

Appropriate correction is required.

Claim Rejections – 35 USC § 103(a)

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made

4. Claims 1-8, 11-24, and 27-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Harsham et al., (Pat. No. 6,041,347) (hereinafter

'Harsham') in view of Cicciarelli et al., (Pub. No. US 2003/0037327 A1) (hereinafter 'Cicciarelli')

5. **As to claim 1**, Harsham discloses a computer-readable medium containing code (Abstract, Lines 3-11, 13-15) which when executed causes a network device to implement a method of managing a configuration (Fig. 4, element 92 – configuration application module; Col. 2, Lines 18-21) of at least a group of nodes (Fig. 4, elements – 26, 28 – machine; Col. 1, Lines 64-66 – simultaneously configuring parameters of devices in a computer network) comprising: an input code (Fig. 4, elements 82 – user input, 84 – rules, groups, network organization), capable of receiving a set of model configuration files (Fig. 4, element 90 – configuration file(s)) adapted to create and configure at least partially a data model (Fig. 4, element 89 – configuration output module; Col. 5, Line 64 through Col. 6 Line 9), wherein the data model defines hardware entities (Col. 1, Lines 50-54 – physical topology; Col. 2, 20-35 – a logic network object group class is a subclass of the abstract network object class; Col. 2, Lines 35-44 – the system also defines a physical network object group class, which is a subclass of the abstract network object class) and logical entities for a group of nodes (Col. 1, Lines 50-54 – logical groups of machines and users in the computer network); a generator code, capable of generating a first node data in cooperation with the at least partially configured data model (Col. 2, Lines 53-57 – means for providing simultaneous configuration of the network devices using the rules defined for each network device in the computer network).

Although Harsham discloses performing simultaneous configuration based on physical/logical topologies rules/model for a group of machines (Abstract, lines 3-11, 13-15), but does not explicitly disclose an install code, capable of installing a specific environment in a machine having at least partially the configuration of the nodes of the group in order to create an archive object using the first node data; configuration code, enabling a user to complete the configuration of the data model dynamically and generate a second node data; and the install code, further capable of installing a specific environment in the group of nodes in order to create a deployable object from the archive object and the second node data and to configure nodes of the group of nodes in deploying the deployable object.

However, in an analogous art of run-time rule-based topological installation suite, Cicciarelli discloses an install code, capable of installing a specific environment in a machine ([0017]; [0018] – dynamically obtaining the topology of the target run-time environment and using this topology information as input to a rules engine for purposes of automatically selecting a particular configuration of an in installation suite) having at least partially the configuration of the nodes of the group in order to create an archive object using the first node data; configuration code, enabling a user to complete the configuration of the data model dynamically and generate a second node data ([0020] – automatically and dynamically adapted for a particular target topology; [0022] – the specified conditions pertain to a target run-time environment and the at least one action may be used to select from among the topologies) and the install

code, further capable of installing a specific environment in the group of nodes in order to create a deployable object from the archive object and the second node data and to configure nodes of the group of nodes in deploying the deployable object ([0023] – the instantiating may further comprise instantiating an object for the particular software installation package and one or more component objects for each software component included in the particular software installation package; [0025], Lines 1-3 – using the populated object model may further comprise: identifying one or more target machines on which the particular software installation package is to be installed).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cicciarelli into the Harsham's system to further provide an install code, capable of installing a specific environment in a machine having at least partially the configuration of the nodes of the group in order to create an archive object using the first node data; configuration code, enabling a user to complete the configuration of the data model dynamically and generate a second node data; and the install code, further capable of installing a specific environment in the group of nodes in order to create a deployable object from the archive object and the second node data and to configure nodes of the group of nodes in deploying the deployable object in Harsham system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Cicciarelli's system which enables installation suites to be more flexible and efficient than prior art static installation suites, by

dynamically obtaining the topology of the target run-time environment and using this topology information as input to rules engines for purposes of automatically selecting a particular configuration of an installation suite as once suggested by Cicciarelli (i.e., [0018]; [0020]).

6. **As to claim 18**, Harsham discloses a method of managing a configuration of at least a group of nodes comprising: receiving a set of model configuration files (Fig. 4, elements 90 – configuration file(s), 92 – configuration application module; Col. 2, Lines 18-21); configuring at least partially a data model as a function of the model configuration files, wherein the data model defines hardware entities (Col. 1, Lines 50-54 – physical topology; Col. 2, 20-35 – a logic network object group class is a subclass of the abstract network object class; Col. 2, Lines 35-44 – the system also defines a physical network object group class, which is a subclass of the abstract network object class) and logical entities (Col. 1, Lines 50-54 – logical groups of machines and users in the computer network) for the group of nodes; generating a first node data as a function of the at least partially configured data model (Fig. 4, element 89 – configuration output module; Col. 5, Line 64 through Col. 6 Line 9).

Although Harsham discloses performing simultaneous configuration based on physical/logical topologies rules/model for a group of machines (Abstract, lines 3-11, 13-15), but does not explicitly disclose installing a specific environment in a machine having at least partially the configuration of the group of nodes in order to create an archive object using the first node data; completing

the configuration of the data model dynamically and generate a second node data; and installing the specific environment in the group of nodes in order to create a deployable object from the archive object and the second node data and in order to configure the nodes of the group of nodes in deploying the deployable object.

However, in an analogous art of run-time rule-based topological installation suite, Cicciarelli discloses installing a specific environment in a machine ([0017]; [0018] – dynamically obtaining the topology of the target run-time environment and using this topology information as input to a rules engine for purposes of automatically selecting a particular configuration of an in installation suite) having at least partially the configuration of the group of nodes in order to create an archive object using the first node data; completing the configuration of the data model dynamically and generate a second node data ([0020] – automatically and dynamically adapted for a particular target topology; [0022] – the specified conditions pertain to a target run-time environment and the at least one action may be used to select from among the topologies); and installing the specific environment in the group of nodes in order to create a deployable object from the archive object and the second node data and in order to configure the nodes of the group of nodes in deploying the deployable object ([0023] – the instantiating may further comprise instantiating an object for the particular software installation package and one or more component objects for each software component included in the particular software installation package; [0025], Lines 1-3 – using the populated object model may further comprise:

identifying one or more target machines on which the particular software installation package is to be installed).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cicciarelli into the Harsham's system to further provide installing a specific environment in a machine having at least partially the configuration of the group of nodes in order to create an archive object using the first node data; completing the configuration of the data model dynamically and generate a second node data; and installing the specific environment in the group of nodes in order to create a deployable object from the archive object and the second node data and in order to configure the nodes of the group of nodes in deploying the deployable object in Harsham system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Cicciarelli's system which enables installation suites to be more flexible and efficient than prior art static installation suites, by dynamically obtaining the topology of the target run-time environment and using this topology information as input to rules engines for purposes of automatically selecting a particular configuration of an installation suite as once suggested by Cicciarelli (i.e., [0018]; [0020]).

7. **As to claims 4 and 19**, Harsham discloses the first node data is stored in a repository enabling the re-creation of the archive object (Col. 8, Lines 53-55).

8. **As to claims 2, 5 and 21**, Harsham does not disclose the second node data is stored in a repository enabling the re-creation of a deployable object.

However, in an analogous art of run-time rule-based topological installation suite, Cicciarelli discloses the second node data is stored in a repository enabling the re-creation of a deployable object (Fig. 9, element 960 – store customization input; [0064] – this information may be obtained in advance and stored for use when the suite installation commences; [0097], Lines 1-13).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cicciarelli into the Harsham's system to further provide the second node data is stored in a repository enabling the re-creation of a deployable object Harsham system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Cicciarelli's system which enables installation suites to be more flexible and efficient than prior art static installation suites, by dynamically obtaining the topology of the target run-time environment and using this topology information as input to rules engines for purposes of automatically selecting a particular configuration of an installation suite as once suggested by Cicciarelli (i.e., [0018]; [0020]).

9. **As to claims 6 and 20**, Harsham discloses the computer-readable medium wherein the first node data comprises software data (Col. 2, Lines 12-14; Col. 11, Lines 3-5)

Harsham does not disclose a software install script for the group of nodes.

However, in an analogous art of run-time rule-based topological installation suite, Cicciarelli discloses a software install script for the group of nodes ([0082], Lines 1-6; [0083]; [0088], Lines 1-4; [0090], Lines 5-10).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cicciarelli into the Harsham's system to further provide a software install script for the group of nodes in Harsham system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Cicciarelli's system which enables installation suites to be more flexible and efficient than prior art static installation suites, by dynamically obtaining the topology of the target run-time environment and using this topology information as input to rules engines for purposes of automatically selecting a particular configuration of an installation suite as once suggested by Cicciarelli (i.e., [0018]; [0020]).

10. **As to claims 3 and 22**, Harsham does not disclose the second node data comprises operating system files and service configuration files for the group of nodes.

However, in an analogous art of run-time rule-based topological installation suite, Cicciarelli discloses the second node data comprises operating system files ([0059] – such as the required operating system; [0065], Lines 10-13 – the operating system(s) installed in the run-time environment may also be used

in the rule predicates to influence the selection o a configuration) and service configuration files ([0064], Lines 3-7; [0083]; [0084]) for the group of nodes.

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cicciarelli into the Harsham's system to further provide the second node data comprises operating system files and service configuration files for the group of nodes in Harsham system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Cicciarelli's system which enables installation suites to be more flexible and efficient than prior art static installation suites, by dynamically obtaining the topology of the target run-time environment and using this topology information as input to rules engines for purposes of automatically selecting a particular configuration of an installation suite as once suggested by Cicciarelli (i.e., [0018]; [0020]).

11. **As to claim 7**, Harsham does not disclose the computer-readable medium wherein the install code is further adapted to create a configured archive object in adding user defined configuration data and user install scripts to the archive object.

However, in an analogous art of run-time rule-based topological installation suite, Cicciarelli discloses the computer-readable medium wherein the install code is further adapted to create a configured archive object ([0022], Line 3 through [0024], Line 10; [0025]; [0026]) in adding user defined configuration

data ([0053]-[0054]) and user install scripts (i.e., [0118], Lines 6-12) to the archive object.

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cicciarelli into the Harsham's system to further provide the computer-readable medium wherein the install code is further adapted to create a configured archive object in adding user defined configuration data and user install scripts to the archive object in Harsham system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Cicciarelli's system which enables installation suites to be more flexible and efficient than prior art static installation suites, by dynamically obtaining the topology of the target run-time environment and using this topology information as input to rules engines for purposes of automatically selecting a particular configuration of an installation suite as once suggested by Cicciarelli (i.e., [0018]; [0020]).

12. **As to claim 8**, Harsham does not disclose the computer-readable medium wherein the install code is further adapted to create a new configured object in adding new user defined configuration data and new user install scripts to an already configured archive object.

However, in an analogous art of run-time rule-based topological installation suite, Cicciarelli discloses the computer-readable medium wherein the install code is further adapted to create a new configured object in adding new

user defined configuration data ([0053]-[0054]) and new user install scripts (i.e., [0118], Lines 6-12) to an already configured archive object ([0022], Line 3 through [0024], Line 10; [0025]; [0026]).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cicciarelli into the Harsham's system to further provide the computer-readable medium wherein the install code is further adapted to create a new configured object in adding new user defined configuration data and new user install scripts to an already configured archive object in Harsham system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Cicciarelli's system which enables installation suites to be more flexible and efficient than prior art static installation suites, by dynamically obtaining the topology of the target run-time environment and using this topology information as input to rules engines for purposes of automatically selecting a particular configuration of an installation suite as once suggested by Cicciarelli (i.e., [0018]; [0020]).

13. **As to claim 11**, Harsham discloses the computer-readable medium wherein the input code is further adapted to receive the set of model configuration files comprising a logical configuration file for the group of nodes and a hardware configuration file, and is further adapted to receive a network configuration file in order to complete the data model (Abstract, Lines 3-11 – these rules may be inherited by machines through both the physical and logical

relationships of the machines in the computer network; Fig. 4, elements 84, 88, 89, 90, 92; Fig. 7A; Col. 2, Lines 27-44).

14. **As to claims 12 and 29**, Harsham discloses the data model as at least partially configured comprises a hardware model linked to a logical model for the group of nodes (Abstract, Lines 3-11 – these rules may be inherited by machines through both the physical and logical relationships of the machines in the computer network; Fig. 4, elements 84, 88, 89, 90, 92).

15. **As to claims 13 and 30**, Harsham discloses the completed data model comprises a hardware model linked to a logical model for the group of nodes and to a network model (Abstract, Lines 3-11; Fig. 4, element 84 – network organization; Fig. 7A; Col. 2, Lines 27-44).

16. **As to claims 14 and 31**, Harsham discloses the data model is organized in classes (Fig. 6A-6D, 8A-8C; Col. 2, Lines 27-44).

17. **As to claim 15 and 32**, Harsham discloses the machine is a prototype machine having at least partially the configuration of nodes of the group of nodes for which an archive object is created (Col. 2, Lines 53-57 – providing simultaneous configuration of all the network devices using the rules defined for each network device in the computer network).

18. **As to claims 16 and 33**, Harsham does not disclose the deployable object is a deployable flash archive.

However, in an analogous art of run-time rule-based topological installation suite, Cicciarelli discloses the deployable object is a deployable flash archive ([0094]).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cicciarelli into the Harsham's system to further provide that the deployable object is a deployable flash archive in Harsham system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Cicciarelli's system which enables installation suites to be more flexible and efficient than prior art static installation suites, by dynamically obtaining the topology of the target run-time environment and using this topology information as input to rules engines for purposes of automatically selecting a particular configuration of an installation suite as once suggested by Cicciarelli (i.e., [0018]; [0020]).

19. **As to claims 17 and 34**, Harsham discloses the archive object is a flash archive (Col. 4, Lines 50-63; Col. 7, Lines 47-51; Col. 8, Lines 53-55).

20. **As to claim 23**, Although Harsham discloses performing simultaneous configuration based on physical/logical topologies rules/model for a group of machines (Abstract, lines 3-11, 13-15), but does not explicitly disclose the

method wherein the installing the specific environment in the machine having at least partially the configuration of the group of nodes further comprises a configured archive object by adding user defined configuration data and user install script to the archive object, the configured archive object being used to create the deployable object.

However, in an analogous art of run-time rule-based topological installation suite, Cicciarelli discloses the method wherein the installing the specific environment in the machine having at least partially the configuration of the group of nodes ([0023] – the instantiating may further comprise instantiating an object for the particular software installation package and one or more component objects for each software component included in the particular software installation package; [0025], Lines 1-3 – using the populated object model may further comprise: identifying one or more target machines on which the particular software installation package is to be installed) further comprises a configured archive object by adding user defined configuration data ([0053]-[0054]) and user install script (i.e., [0118], Lines 6-12) to the archive object, the configured archive object ([0022], Line 3 through [0024], Line 10; [0025]; [0026]) being used to create the deployable object (Figs. 10-11; [0102]-[0103]).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cicciarelli into the Harsham's system to further provide the method wherein the installing the specific environment in the machine having at least partially the configuration of the group of nodes further comprises a configured archive object by adding user

defined configuration data and user install script to the archive object, the configured archive object being used to create the deployable object in Harsham system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Cicciarelli's system which enables installation suites to be more flexible and efficient than prior art static installation suites, by dynamically obtaining the topology of the target run-time environment and using this topology information as input to rules engines for purposes of automatically selecting a particular configuration of an installation suite as once suggested by Cicciarelli (i.e., [0018]; [0020]).

21. **As to claim 24**, Harsham does not disclose the method wherein the installing the specific environment in the machine having at least partially the configuration of the group of nodes further comprises creating a new configured object by adding new user defined configuration data and new user install scripts to the already configured archive object.

However, in an analogous art of run-time rule-based topological installation suite, Cicciarelli discloses the method wherein the installing the specific environment in the machine having at least partially the configuration of the group of nodes further comprises creating a new configured object by adding new user defined configuration data ([0053]-[0054]) and new user install scripts (i.e., [0118], Lines 6-12) to the already configured archive object ([0022], Line 3 through [0024], Line 10; [0025]; [0026]).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cicciarelli into the Harsham's system to further provide the method wherein the installing the specific environment in the machine having at least partially the configuration of the group of nodes further comprises creating a new configured object by adding new user defined configuration data and new user install scripts to the already configured archive object in Harsham system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Cicciarelli's system which enables installation suites to be more flexible and efficient than prior art static installation suites, by dynamically obtaining the topology of the target run-time environment and using this topology information as input to rules engines for purposes of automatically selecting a particular configuration of an installation suite as once suggested by Cicciarelli (i.e., [0018]; [0020]).

22. **As to claim 27**, Harsham discloses the method wherein the set of model configuration files (Fig. 4, element 90 – configuration file(s)) comprises a logical configuration file for the group of nodes (Col. 1, Lines 50-54 – logical groups of machines and users in the computer network) and a hardware configuration file (Col. 1, Lines 50-54 – physical topology; Col. 2, 20-35 – a logic network object group class is a subclass of the abstract network object class; Col. 2, Lines 35-44 – the system also defines a physical network object group class, which is a subclass of the abstract network object class).

23. **As to claim 28,** Harsham discloses the method further comprising receiving a network configuration file (Col. 2, Lines 29-42 – an abstract network object class: a logical network object group class and a physical network object group class); and completing configuration the data model as a function of the network configuration file (Fig. 5A – hierarchal view; Fig. 4, elements 84 – network organization, 89 – configuration output module, 90 – configuration file(s), 92 – configuration application module, 96 – configuration file; Col. 5, Line 64 through Col. 6, Lines 18).

24. Claims 9-10 and 25-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Harsham, as applied in claim 7 above, in view of Cicciarelli, as applied in claim 7 above, and in further view of Burkhardt et al., (Pub. No. US 2003/0037326 A1) (hereinafter 'Burkhardt')

25. **As to claim 9**, incorporating the rejection in claim 7, Harsham and Cicciarelli do not disclose the computer-readable medium wherein the install code is further adapted to configure nodes of the group of nodes, to reboot the nodes, to configure the user defined configuration data and to run the nodes.

However, in an analogous art of method and system for installing staged programs on a destination computer using a reference system image, Burkhardt discloses the computer-readable medium wherein the install code is further adapted to configure nodes of the group of nodes, to reboot the nodes, to

configure the user defined configuration data and to run the nodes ([0032], Lines 11-19 – on each boot until ... it has reached the last configuration step, ... including the operating system, is re-configured).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Burkhardt into the Harsham- Cicciarelli's system to further provide the computer-readable medium wherein the install code is further adapted to configure nodes of the group of nodes, to reboot the nodes, to configure the user defined configuration data and to run the nodes object in Harsham-Cicciarelli system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Burkhardt's system that advantageously provides a dramatic reduction in the time it takes to produce customized machines, provides significant cost savings in maintaining the reference system image, and facilitates updating the operating system and other programs as once suggested by Burkhardt (i.e., [0005]).

26. **As to claim 10**, incorporating the rejection in claim 7, Harsham and Cicciarelli do not disclose the computer-readable medium wherein the install code is further adapted to configure nodes of the group of nodes, to perform a first reboot of the nodes, to configure the user defined configuration data, to perform a second reboot of the nodes and to run the nodes.

However, in an analogous art of method and system for installing staged programs on a destination computer using a reference system image, Burkhardt

discloses the computer-readable medium wherein the install code is further adapted to configure nodes of the group of nodes, to perform a first reboot of the nodes, to configure the user defined configuration data, to perform a second reboot of the nodes and to run the nodes ([0032], Lines 11-19 – on each boot until ... it has reached the last configuration step, ... including the operating system, is re-configured).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Burkhardt into the Harsham- Cicciarelli's system to further provide the computer-readable medium wherein the install code is further adapted to configure nodes of the group of nodes, to perform a first reboot of the nodes, to configure the user defined configuration data, to perform a second reboot of the nodes and to run the nodes in Harsham-Cicciarelli system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Burkhardt's system that advantageously provides a dramatic reduction in the time it takes to produce customized machines, provides significant cost savings in maintaining the reference system image, and facilitates updating the operating system and other programs as once suggested by Burkhardt (i.e., [0005]).

27. **As to claim 25**, incorporating the rejection in claim 23, Harsham and Cicciarelli do not disclose the method wherein installing the specific environment in the group of nodes further comprises configuring nodes of the group of nodes,

rebooting the nodes, configuring the user defined configuration data and running the nodes.

However, in an analogous art of method and system for installing staged programs on a destination computer using a reference system image, Burkhardt discloses the method wherein installing the specific environment in the group of nodes further comprises configuring nodes of the group of nodes, rebooting the nodes, configuring the user defined configuration data and running the nodes ([0032], Lines 11-19 – on each boot until ... it has reached the last configuration step, ... including the operating system, is re-configured).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Burkhardt into the Harsham- Cicciarelli's system to further provide the method wherein installing the specific environment in the group of nodes further comprises configuring nodes of the group of nodes, rebooting the nodes, configuring the user defined configuration data and running the nodes in Harsham-Cicciarelli system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Burkhardt's system that advantageously provides a dramatic reduction in the time it takes to produce customized machines, provides significant cost savings in maintaining the reference system image, and facilitates updating the operating system and other programs as once suggested by Burkhardt (i.e., [0005]).

28. **As to claim 26**, incorporating the rejection in claim 23, Harsham and Cicciarelli do not disclose the method wherein installing the specific environment in the group of nodes further comprises configuring nodes of the group of nodes, performing a first reboot of the nodes, configuring the user defined configuration data, performing a second reboot of the nodes and running the nodes.

However, in an analogous art of method and system for installing staged programs on a destination computer using a reference system image, Burkhardt discloses the method wherein installing the specific environment in the group of nodes further comprises configuring nodes of the group of nodes, performing a first reboot of the nodes, configuring the user defined configuration data, performing a second reboot of the nodes and running the nodes ([0032], Lines 11-19 – on each boot until … it has reached the last configuration step, … including the operating system, is re-configured).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Burkhardt into the Harsham- Cicciarelli's system to further provide the method wherein installing the specific environment in the group of nodes further comprises configuring nodes of the group of nodes, performing a first reboot of the nodes, configuring the user defined configuration data, performing a second reboot of the nodes and running the nodes in Harsham-Cicciarelli system.

The motivation is that it would enhance the Harsham's system by taking, advancing and/or incorporating Burkhardt's system that advantageously provides a dramatic reduction in the time it takes to produce customized machines,

provides significant cost savings in maintaining the reference system image, and facilitates updating the operating system and other programs as once suggested by Burkhardt (i.e., [0005]).

Conclusion

29. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

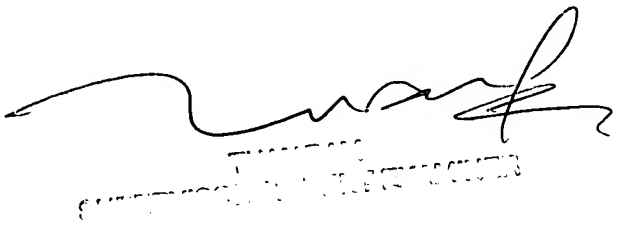
- P. Hanchett, *System and Method for Configuration, Management, and Monitoring of a Computer Network Using Inheritance* (Pat. No. US 6,834,301 B1)
- G. C. Hunt, *Reversible Load-Time Dynamic Linking* (Pat. No. US 6,499,137 B1)
- Cicciarelli et al., *Efficient Installation of Software Packages* (Pub. No. US 2002/0188941 A1)
- Guilbeault et al., *Apparatus for Managing the Installation of Software Across a Network* (Pat. No. US 6,883,169 B1)

30. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ben C. Wang whose telephone number is 571-270-1240. The examiner can normally be reached on Monday - Friday, 8:00 a.m. - 5:00 p.m., EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on 571-272-3695. The fax

phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.



BCW BW

March 14, 2007